RENESAS Tool News

The C/C++ Compiler Packages for the H8, H8S, and H8SX MCU Families Revised to V.6.01 Release 01

The C/C++ compiler packages for the H8, H8S, and H8SX MCU families have been revised from V.6.01 Release 00 to V.6.01 Release 01.

1. **Products Concerned**

    The H8, H8S, and H8SX MCU families C/C++ compiler packages V.6.
    (Products for Windows, Solaris, and HP-UX concerned.)

2. **Descriptions of Revision**

    2.1   Functions Introduced and Improved

    2.1.1   In the High-performance Embedded Workshop (Windows Version Only)
    The High-performance Embedded Workshop (an Integrated Development Environment) bundled with the package has been revised to V.4.00.02.
    For details, see RENESAS TOOL NEWS "The High-performance Embedded Workshop, an Integrated Development Environment, Revised to V.4.00.02" (No. RSO-HEW-050701D), issued on July 1, 2005.

    2.1.2   In the Simulator (Windows Version Only)
    The timer simulation function has been separated from the simulator body to be executed as an I/O DLL.

2.2   Problems Fixed
       The problems described in Sections 2.2.1 through 2.2.4 below have
       been fixed.

   2.2.1   In the Installer (Windows Version Only)
            On executing the batch file for setting environmental
            variables
            For details, see RENESAS TOOL NEWS "A Note on Using
            the C/C++ Compiler Packages for the SuperH RISC Engine;
            and the H8, H8S and H8SX Families (Windows Versions
            Only)" (No. RSO-SHC_1-050316D), issued on March 16,
            2005.

   2.2.2   In the Simulator (Windows Version Only)
            (1)   On using the simulator debugger for the SuperH
                   RISC engine family and the one for the H8, H8S and
                   H8SX families in the same host system
                   For details, see RENESAS TOOL NEWS "A Note on
                   Using the Simulator Debuggers for the SuperH RISC
                   Engine; and the H8, H8S, and H8SX Families" (No.
                   RSO-SHC_2-050316D), issued on March 16, 2005.

            (2)   On setting access to memory resources
                   For details, see RENESAS TOOL NEWS "A Note on
                   Using the Simulator Debugger for the SuperH RISC
                   Engine; and for the H8S Families and the H8/300
                   Series of MCUs--On Setting Access to Memory
                   Resources" (No. RSO-SH-SIM-050416D), issued on
                   April 16, 2005.

            (3)   On accessing system control register SYSCR
                   For details, see RENESAS TOOL NEWS "A Note on
                   Using the Simulator Debugger for the H8S Family
                   and H8/300 Series of MCUs--On Accessing System
                   Control Register SYSCR" (No. RSO-H8SX-SIM-
                   050416D), issued on April 16, 2005.

            (4)   On searching for trace records
                   For details, see RENESAS TOOL NEWS "A Note on
                   Using the Simulator Debugger for the H8S Family
                   and H8/300 Series of MCUs--On Searching for
                   Trace Records" (No. RSO-H8S-SIM-050416D),
                   issued on April 16, 2005,

2.2.3   In the Compilers

(1)   Eight problems reported in RENESAS TOOL NEWS "Notes on Using the C/C++ Compiler Package V.6 for the H8SX, H8S, and H8 Families of MCUs" (No. RSO-H8C-050716D), issued on July 16, 2005.

(2)   On referencing incorrect memory area
For details, see RENESAS TOOL NEWS "A Note on Using the C/C++ Compiler Package V.6 for the H8SX, H8S, and H8 Families" (No. RSO-H8C-050801D), issued on August 1, 2005.

(3)   Also the following four problems resolved:
- On incorrectly displaying information on the register that contains the values of variables and union-type ones on the debugger because the compiler generates incorrect debug information
- On displaying errors C2190, C2240, C4712, and C2238 accidentally
- On not displaying error C2311 when it arises
- On displaying assemble errors (W)835 or (E)526 when assembly programs generated by the compiler are assembled

2.2.4   In the Linker (Optimizing Linkage Editor optlnk)

(1)   On linkage list files generated by the optimizing linkage editor
For details, see RENESAS TOOL NEWS "A Note on Using the Optimizing Linkage Editor for the SuperH RISC Engine, H8, H8S, and H8SX Families of MCUs" (No. RSO-SHC-050416D), issued on April 16, 2005.

(2)   On displaying the error message shown below accidentally
** L0103 (I) Multiple stack sizes specified to the symbol "Function Name"
Conditions:
This problem may occur if the following conditions are all satisfied:
1. The version of the linker is V.9.00.00 or later.*
2. An extern-qualified variable that has the same

name as a function defined in a C/C++ source file exists in another C/C++ source file. Or, an import label of "_<Function Name>" exists in an assembly source file.

   3. The message option is used at linking.

   4. Also the stack option is used at linking.

(3)   On generating an incorrect stack information file (.sni) Conditions:

This problem may occur if the following conditions are all satisfied:

   1. The version of the linker is V.9.00.00 or later.*

   2. An extern-qualified variable that has the same name as a function defined in a C/C++ source file exists in another C/C++ source file. Or, an import label of "_<Function Name>" exists in an assembly source file.

   3. The stack option is used at linking.

NOTICE:

If an incorrect .sni file is read into the stack analyzing tool, Call Walker, incorrect stack usage is provided.

(4)   On generating incorrect object code by using the "optimize=symbol _delete" option to optimize the deletion of unreferenced symbols

Conditions:

This problem may occur if the following conditions are all satisfied:

   1. The version of the linker is V.9.00.00 or later.*

   2. The optimize option is used at compilation.

   3. Also the pack=1 option is used at compilation, or "#pragma pack 1" is put in a C/C++ source file.

   4. Optimization of the deletion of unreferenced symbols is valid in the linker.

     Optimization becomes valid in any of the following cases:

       - The optimize=symbol_delete option is used.
       - The optimize=speed option is used.
       - The optimize option is used.
       - The nooptimize option is not used.

5. Constants (const- qualified variables) and variables having their initial values are deleted as unreferenced variables by the optimization in Condition 4.

(5) On generating incorrect debug information by using the compress option to compress debug information
Conditions:
This problem may occur if the following conditions are all satisfied:
1. The version of the linker is V.9.00.00 or later.*
2. The "debug" option is used at compilation.
3. Also the "compress" option is used for generating relocatable files (.rel) at linking.
4. The relocatable files in Condition 3 are read into the linker to generate a load module using the "compress" option.
NOTICE:
Loading the above load module in the debugger results in errors.

(6) Also the problem that internal errors arise if any of the following conditions is met is resolved:
   ○ The linker reads .rel files written in the assembly language and containing .EQU symbols (internal error L4000-8010).
   ○ The linker reads object files containing function calls calling .EQU symbols (internal error L4000-8874).
   ○ The linker reads .rel files (internal error L4000-8027 or L4001).

* How to check for the version number of your linker

(1) In the High-performance Embedded Workshop, open the Tool menu and select the Administration command. The Tool Administration dialog box appears.

(2) Out of the Toolchains tree in the Registered Components list, select the name of the compiler package you are using; then click the properties button.

(3) Click the Information tab in the Properties dialog box, and you see the version number of your linker. Example of Display: Optimizing Linkage Editor

(V.9.00.02)

3. How to Update Your Package and Purchase the Revised One

   3.1 Free-of-Charge Update
   Free-of-charge update is available if you are using the
   product concerned.

   (1)   Windows version
         To update yours online, download the revised
         product from **HERE**.

   (2)   Solaris and HP-UX versions
         Supply the following items of information to your
         local Renesas Technology sales office or distributor.
         We will send you the latest version of the product
         package by return:

Type Name	Solaris or HP-UX version
Version No.	V.6.01
Release No.	Release 01

   3.2 First Ordering
   If you place an order for any of the products, please supply
   the following items of information to your local Renesas
   Technology sales office or distributor:

Type Name	Windows, Solaris, or HP-UX version
Version No.	V.6.01
Release No.	Release 01